A VITERBI DECODER SUITABLE FOR VLSI IMPLEMENTATION

A Thesis
Submitted to the Faculty of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree of
Master of Science
in the Department of Electrical Engineering
University of Saskatchewan
by

Brian Lyle Frank Daku
Saskatoon, Saskatchewan

SEPTEMBER, 1986

Copyright

Acknowledgements

The author wishes to thank Professors D.E. Dodds and A.G. Wacker for their guidance throughout the course of this work. Their advice and assistance in the preparation of this thesis is appreciated. He also thanks his wife, Jan, and son Jonathon, for their patience and encouragement.

The author also extends thanks to Professor R.J. Bolton for his assistance in the VLSI portion of the project and Professor V.K. Bhargava of the University of Victoria for his assistance in the area of error control coding.

Financial assistance, in the form of scholarships, was provided by SED Systems Inc. and the Natural Science and Engineering Research Council (IPGS). Their support is gratefully acknowledged.

UNIVERSITY OF SASKATCHEWAN

Electrical Engineering Abstract 86A271

A VITERBI DECODER SUITABLE FOR VLSI IMPLEMENTATION

Student: B.L. Daku
Supervisors: D.E. Dodds and A.G. Wacker
M.Sc. Thesis presented to the College of Graduate Studies

September 1986

Abstract

There appears to be a market for low cost, low data rate private satellite communication networks. These networks require a low cost digital earth station to make them economically viable. One method to lower the cost of the earth station is to use an error control scheme to reduce the power required at the expense of increased bandwidth. This thesis examines forward error control (FEC) techniques that can be used in this environment.

The main objective of this project is to determine if it is feasible to design a suitable FEC system that can be implemented on a single integrated circuit. This FEC system must handle an information rate of at least 56 Kbps and have a coding gain of greater than 5 dB at a decoded bit error rate of 10^{-6}.

A major portion of the thesis is dedicated to building a suitable information base that can be used to evaluate possible implementations. The thesis presents a possible implementation architecture for a single chip Viterbi FEC system. A portion of the Viterbi decoder circuit is implemented in VLSI and simulated using SPICE to determine the maximum information rate. An estimate is made of the coding gain and the required size of the integrated circuit.

The thesis concludes that it is possible to implement a suitable Viterbi FEC system on a single integrated circuit.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ACS	Add-Compare-Select
ARQ	Automatic Repeat Request
AWGN	Additive White Gaussian Noise
BER	Bit Error Rate
BMV	Branch Metric Value
BPSK	Binary Phase Shift Keying
BSC	Binary Symmetric Channel
CLA	Carry Look Ahead
CMOS	Complimentary Metal Oxide Semiconductor
DMC	Discrete Memoryless Channel
FEC	Forward Error Control
IC	Integrated Circuit
Kbps	Kilobits per second
LSB	Least Significant Bit
MSB	Most Significant Bit
PMR	Path Metric Range
PMV	Path Metric Value
PSK	Phase Shift Keying
QPSK	Quadrature Phase Shift Keying

RAM	Random Access Memory
ROM	Read Only Memory
SNR	Signal to Noise Ratio
VLSI	Very Large Scale Integration
XOR	Exclusive OR

Chapter 1

INTRODUCTION

1.1. Background

The deregulation of the telecommunication industry is providing an exceptional opportunity for the establishment of private satellite communication networks. There has been much activity, to date, in high cost, high data rate satellite communications. These systems have typically been used by large companies, with the aim of reducing their terrestrial telecommunications costs. There also appears to be a market for low cost, low data rate private satellite communication networks. Products to supply this market are now under development. The main terrestrial component for these small satellite data networks is a low cost digital earth station. It is necessary to minimize the cost of this digital earth station to make these networks economically viable. One significant way to reduce the cost is to use an error control scheme.

When digital data is sent over a noisy channel, there is a finite probability that the received signals will not exactly match the transmitted signals. This may result in errors that render the received data unusable. Typically, the user establishes a maximum error rate for acceptable data transmission. In a communication system that will not meet the error rate requirements, error control coding is one method that can be used to reduce the number of errors to this maximum level, at a cost of increased bandwidth. These errors could also be reduced by increasing the transmitted power or antenna size, but in the satellite communications environment these approaches are more expensive than error control coding.

Error control coding can best be introduced using error detection examples that are familiar to most electrical engineers. A very simple method of error detection involves appending a single parity bit to a block of data (codeword). The parity bit is chosen to make the number of '1' bits in the codeword either always even or always odd depending on whether the scheme is based on even or odd parity. This parity bit can be used to detect single errors because any one bit error produces a codeword with incorrect parity. Another error detection scheme that is in widespread use is the cyclic redundancy code or CRC code. This is a scheme that has a relatively simple hardware implementation that can be used to detect a variety of errors. For example, CRC-16 can detect all single and double errors in a codeword, all errors with an odd number of bit inversions, all single burst errors of length 16 bits or less, 99.997% ($1-2^{15}$) of 17 bit error bursts, and 99.998% ($1-2^{16}$) of 18 bit and longer error bursts (or several shorter bursts) [25].

The previous two examples involved only error detection. The proposed low cost earth station requires the use of error correction. When choosing a system for error correction there are three different techniques that can be used: forward-error-control (FEC), error detection with retransmission (automatic repeat request (ARQ)), and hybrid systems that use a combination of FEC and ARQ. This thesis deals specifically with satellite communication systems and thus it is necessary to pick an error control system that is suitable for this environment. The ARQ and hybrid systems require a reverse channel to request retransmission of incorrectly received data. This reverse channel is a costly item to implement in a satellite system. Requests for retransmission will also significantly increase the transmission delay of the data because of the round trip time of approximately 0.27 seconds for the typical case of a satellite in geostationary orbit. For example, if only one repeat request is required, it will take at least 0.81 seconds to transmit the data. In most cases this delay is unacceptable. FEC systems do not require a reverse channel and thus these

systems appear to be the best choice for error control in satellite communications.

The usefulness of coding was originally demonstrated by Shannon in 1948 [22]. He proved that if the information transmission rate is less than a maximum quantity called the channel capacity, then communication over a noisy channel with an error probability as small as desired is possible with proper encoding and decoding. Shannon's work essentially states that available bandwidth, signal power and channel noise set a limit only on the transmission rate and not on accuracy.

There are two common ingredients in all forward error correction schemes, redundancy and noise averaging [4]. Redundancy involves adding extra data symbols to the message based on some encoding algorithm. Thus, although a noisy channel may cause errors in both information and redundant symbols, there is usually still enough information available to the receiver to allow a sophisticated decoder to correct all the errors unless the noise is very severe. Noise averaging involves making the added redundant symbols depend on a span of many information symbols. For example, if the channel has an error rate of P_e, then the fraction of symbols that are in error in a codeword will approach P_e as the codeword length goes to ∞. Thus the longer the codeword, the less likely the fraction of errors in the codeword will exceed P_e. This permits an appropriate coding scheme to decode more codewords correctly.

Historically, the cost of implementing FEC systems has been very high. Initially the high cost of coding limited its use to satellite and space communications, because satellite signal power was very expensive. The introduction of progressively cheaper integrated circuits has reduced the cost of digital hardware significantly and now error control coding has found applications in many areas of communications and memory storage systems. The problem with discrete IC designs is that the decoding algorithms that can be economically implemented is limited. It is now possible to replace

these discrete designs with application specific IC's. This provides an opportunity for designers to implement very complex decoding algorithms at a relatively low cost.

1.2. Objectives

The main objective of this project is to determine if it is feasible to design a FEC system that can be implemented on a single integrated circuit. The FEC system must meet the following constraints:

1. The IC must handle an information rate of $\geq$ 56 Kbps.
2. The coding gain of the system must be $\geq$ 5 dB at a decoded bit error rate (BER) of 10^{-6}. (coding gain is defined in Section 2.2.)

There are many possible implementation aspects that can effect these two constraints. Thus a major portion of this thesis is dedicated to building a suitable information base that can be used by the reader to evaluate the possible implementations.

There have been a number of FEC codecs (coder-decoder) implemented on integrated circuits [5, 20]. None of these implementations meet the above constraints. The closest one is a chip being produced by Stanford Telecommunications [5]. It meets the coding gain requirements, but the maximum information rate is only 9.6 Kbps. It also requires two external memories.

1.3. Project Constraints

In an effort to focus the work of choosing an FEC system, a number of project constraints have been generated. The following constraints have been selected because they reflect a practical industry problem. Some of the terms used in the constraints may be unfamiliar to the reader. These terms will be defined later in the thesis.

1. QPSK (Quadrature Phase-Shift Keying) is chosen as the

modulation scheme because it is presently the main modulation method used in satellite communications.

2. The satellite channel noise is modeled as Additive White Gaussian Noise (AWGN). This is because the dominant degradation is due to additive thermal or galactic noise. [17]

3. A code rate of r = 1/2 is chosen to keep a good balance between the complexity of the circuit and the required coding gain.

4. A convolutional encoder is used to encode the data. This type of coding is conventionally used with long serial data sequences.

1.4. Method

This section describes the method of attack used to verify that the project objectives can be met. It also gives the corresponding chapters in the thesis document. The first step was to determine how error control coding interfaced with the digital communication system and how the coding performance was measured within this system. This information is presented in Chapter 2 of the thesis. The next step was to investigate the options available for convolutional encoding and decoding. Chapter 3 presents the structure of convolutional encoding and the various algorithms available for decoding. It also selects the Viterbi algorithm for implementation. The various design options for the Viterbi decoder were then generated. This was done to provide other designers with the basic information necessary to design or modify Viterbi decoders. To verify the objectives, a specific implementation architecture was selected from the various design options. The Viterbi decoder design options and the implementation architecture selected is presented in Chapter 4. The information rate objective was checked by implementing a time critical portion of the Viterbi decoder. This portion is called the Add-Compare-Select (ACS) circuit. The Add-Compare-Select VLSI layout and propagation delay simulations are presented in Chapter 5. The performance of the Viterbi decoder implementation with respect to the project objectives is discussed in Chapter 6. The thesis conclusions are presented in Chapter 7.

Chapter 2

CODING IN DIGITAL COMMUNICATIONS

Coding is an integral part of many digital communication systems, with satellite communications being just one application. It is worthwhile to put coding into perspective by examining the organization of a typical digital communication system. Section 2.1 does this by describing each major component of a digital communication system. Once the reader understands how coding fits into digital communications, he must become familiar with the methods used to evaluate the performance of coding. Section 2.2 presents the various performance measures that are used in the coding domain. Finally, coding is not applicable to all digital communication systems. Section 2.3 gives a theoretical look at why coding is limited to certain communication channels.

Generally, coding refers to the encoding and decoding of either block or convolutional codes. In this chapter explanations are usually given in terms of both codes even though the Project Constraint section of Chapter 1 eliminated block codes from consideration. Block codes are considered in this introductory chapter in an effort to provide a more general background to coding.

2.1. Description of a Digital Communication System

A digital communication system, broken into it's basic components, is shown in Figure 2-1. This block diagram shows the one way transmission of information from the source to the destination. The following sections describe each of the blocks shown in Figure 2-1.

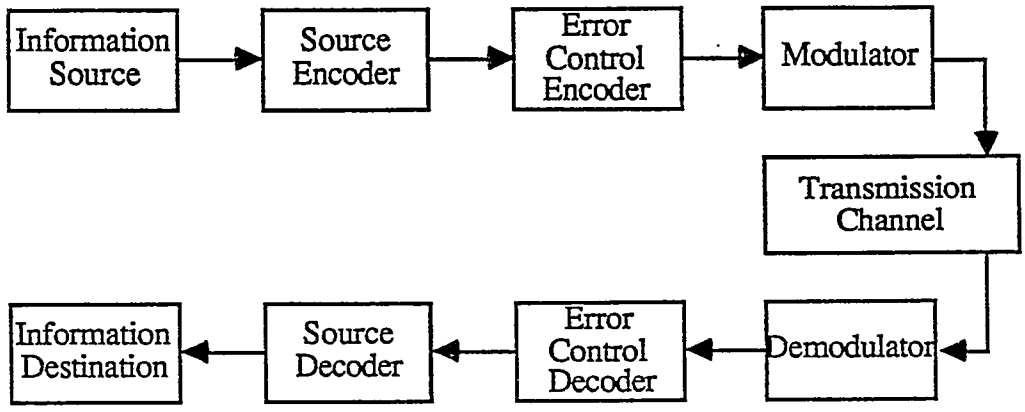

Figure 2-1: Digital Communication System

2.1.1. Information Source / Source Encoder

The Information Source can produce either analog or discrete data. If the source produces analog information, then a source encoder is used to convert this information into digital messages.

The job of the source encoder is to sample the analog source, then use an analog-to-digital converter to produce a digital message. Source encoding also involves data compression to increase the efficiency of the digital communication system. This involves removing as much redundancy as possible from the source information. A typical example of a source encoding process is differential pulse code modulation. This scheme involves using the difference between successive sample values to represent the digital message.

This thesis is not concerned with source encoding and therefore it is assumed the information source produces digital symbols. Unless stated otherwise, it is assumed that the source generates stochastically independent binary symbols, with equal a priori probabilities. These binary digits are produced at a constant rate of R bits per second. R is called the information rate of the source.

2.1.2. Error Control Encoder

The error control encoder manipulates the source data prior to modulation. Error control coding improves performance by inserting redundant algorithmic data into the source information. This method of encoding permits the decoder to detect and correct transmission errors.

This thesis concentrates on error control coding. The following passages briefly describe the available encoding techniques, which are based on block and convolutional codes.

The encoder accepts information at rate R bits per second. It then adds the redundant symbols to produce the encoded data at a higher rate of R_s bits per second.

The encoder for a block code divides the information sequence into blocks of k information bits, a binary k-tuple called a message. There are 2^k different possible messages. The encoder maps each message independently into a binary n-tuple which is called a code word. Thus there are 2^k different possible code words corresponding to the 2^k different possible messages. This set of 2^k code words of length n is called an *(n,k)* block code [4]. The code rate, which can be interpreted as the number of information bits entering the encoder per transmitted symbol, is given by $r = k/n$. Practical values of k range from 3 to several hundred. Practical values of r range from 1/4 to 7/8 [4]. The input and output sequences are usually binary symbols, but they could be symbols from some higher order alphabet.

The encoder for a block code is a memoryless device because each n-symbol output code word depends only on a specific k-bit input message and no others. Since the encoder is memoryless it can be implemented with a combinational logic circuit.

A convolutional encoder operates on long sequences of information symbols

which are encoded continuously in a serial form. The encoding operation can be thought of as the discrete-time convolution of the input sequence with the impulse response of the encoder. In the literature, a convolutional encoder is typically described using a shift register and modulo-2 adders as shown in Figure 2-2 [13].

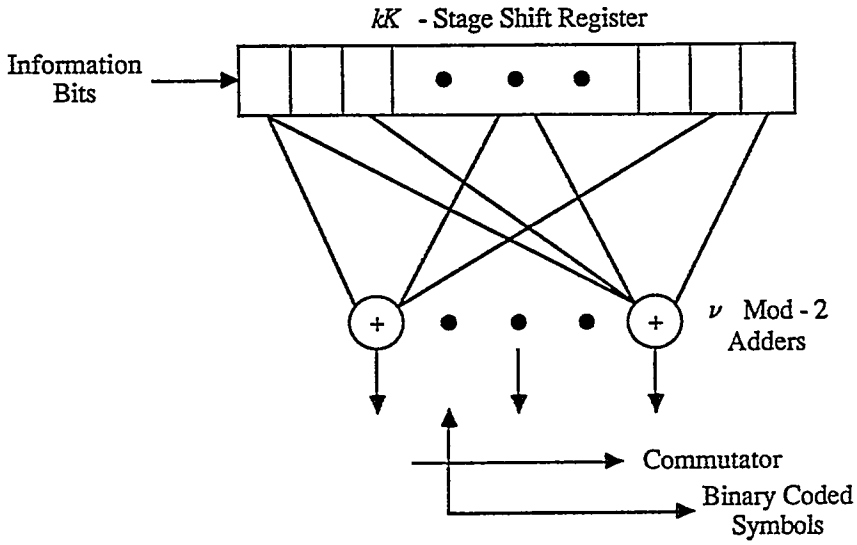

Figure 2-2: Rate k/ν Convolutional Encoder

The binary encoder consists of a $k \cdot K$ stage binary shift register and ν modulo-2 adders. Each of the mod-2 adders is connected to certain stages of the shift register. The specific code used determines these interconnections. The information sequence is shifted into the shift register k bits at a time (k is typically chosen to be 1). Following each k bit shift, each of the mod-2 adders are sampled sequentially to yield the ν bit code word. Since there are ν output bits for k input bits the code rate is given by $r = k/\nu$. K, which is defined as the constraint length of the code, is the number of k bit shifts over which a single bit can influence the encoder output. Practical parameters for convolutional codes are k and ν in the range of 1 to 8, r in the range of 1/4 to 7/8, and K in the range of 2 to 60 [4]. (though most decoding algorithms restrict K to $\leq$ 10 for implementation).

The convolutional encoder is a device with memory of order *K-1*. This memory results because each set of ν output symbols is determined by the current input set of k symbols and by a span of $k(K-1)$ of the preceding input symbols. Since the encoder contains memory it must be implemented with a sequential logic circuit.

2.1.3. Modulator

The function of the modulator is to generate a set of finite time duration analog waveforms. The error control encoder output is mapped by the modulator into this set of waveforms. For binary modulation schemes, each encoder output symbol, 0 or 1, is converted to one of two possible waveforms of equal duration T_s. For M-ary modulation schemes, the encoder output is divided into sets of g symbols ($M=2^g$). Each set is then used to select one of the M waveforms. Conventional modulation schemes for binary operation include phase shift keying (PSK), differentially encoded PSK (DPSK) and frequency shift keying (FSK). There are also M-ary forms of these modulation schemes: MPSK, MDPSK and MFSK [34].

2.1.4. Transmission Channel

The transmission channel consists of all the hardware required to prepare the baseband modulated waveforms for transmission, the physical transmission media, and the hardware required to prepare the received signals for demodulation. Note that the transmission channel does not have to be a communication system, it could also be a data storage or recording system.

In the usual case the output of the channel, $\hat{s}(t)$, is an attenuated version of the input, $s(t)$, to which some random noise, $n(t)$ has been added. This noise makes it difficult for the demodulator to distinguish one modulated waveform from another, resulting in errors and degraded performance. There are many sources of noise, for example thermal noise in electrical circuitry, atmospheric impulse noise such as lightning and burst noise due to signal fading.

Thermal noise is broadband and it has a Guassian amplitude distribution. The errors tend to occur independently from one signalling interval to the next. Impulsive noise is characterized by short periods of intense noise. In this case the output errors occur in bursts. Assuming no fading, the satellite channel noise can be modelled as Gaussian. Channels corrupted with Gaussian noise must be handled using different error control coding techniques than those channels corrupted by impulse noise.

2.1.5. Demodulator

The demodulator estimates which of the possible symbols was transmitted based upon the received signal $\hat{s}(t)$. The probability that this estimate is correct depends upon the signal to noise ratio *(SNR)*, and the detection scheme used. The performance measure used to indicate the average rate of occurrence of symbol errors, taken as the fraction of symbol errors to symbols received over a period of time, is called the probability of symbol error. For binary transmission it is called the bit error rate *(BER)* or the probability of bit error [1]. In a system where no error control coding is used the demodulator output error rate is the error rate of the data delivered to the user. The demodulator output error rate in a coded system is often called the raw channel error rate or the uncoded error rate.

A useful model for the demodulator is an analog filter followed by a quantizer [17]. In the simplest case, for binary transmission the quantizer makes a definite decision for each received symbol, 0 or 1. This is typically called hard decision demodulation and the quantizer has a two level output. If the demodulator incorporates a quantizer with more than two levels, it is referred to as soft decision demodulation. These extra levels can be thought of as a rough measure of how closely a received symbol resembles a '1' or a '0' in the binary case. Soft decision demodulation is used to preserve information that can be used later by the error control decoder to increase the error correction performance. (An approximate performance gain of 2 dB is achieved with eight level soft decision over hard decision.)

2.1.6. Error Control Decoder

The error control decoder is a device which attempts to reverse the operations of the error control encoder. The decoder has a much more difficult task relative to the encoder, because the transmitted data may contain errors. For block codes the decoder repeatedly converts blocks of n sequential data symbols into k information symbols. For convolutional codes, the decoder operates on a continuous stream of data symbols. The data symbols to be decoded can be viewed as being within a sliding window, which makes up the present received symbols and a number of previous symbols. After each ν symbol shift of the sliding window the decoder outputs k decoded information symbols [17].

The inputs to a decoder can be hard decision symbols or soft decision symbols. The decoding techniques used to decode hard decision inputs often involve algebraic equation solving algorithms. The decoding techniques for soft decision inputs resemble signal correlation operations. The advantage of soft decision decoding algorithms, is that for most codes they will out perform the best hard decision decoding algorithm. The use of soft decisions in convolutional codes is relatively straightforward. For block codes the use of soft decision is much more involved.

2.1.7. Source Decoder

The source decoder attempts to reproduce the information originally generated by the analog source. This, of course, is only an estimate of the original information because of the potential errors through the system. Source coding can be viewed as an operation that removes the natural and usually inefficient redundancy from the source. A good error control code inserts a more efficient redundancy to improve performance.

2.1.8. Useful Channel Models

From the viewpoint of the coding, the communication system shown in Figure 2-1 can be simplified. This is done by modeling the modulator, transmission channel and demodulator by a discrete data channel, indicated by the box in Figure 2-3 [17, 4]. The source encoder and decoder have also been removed in the figure.

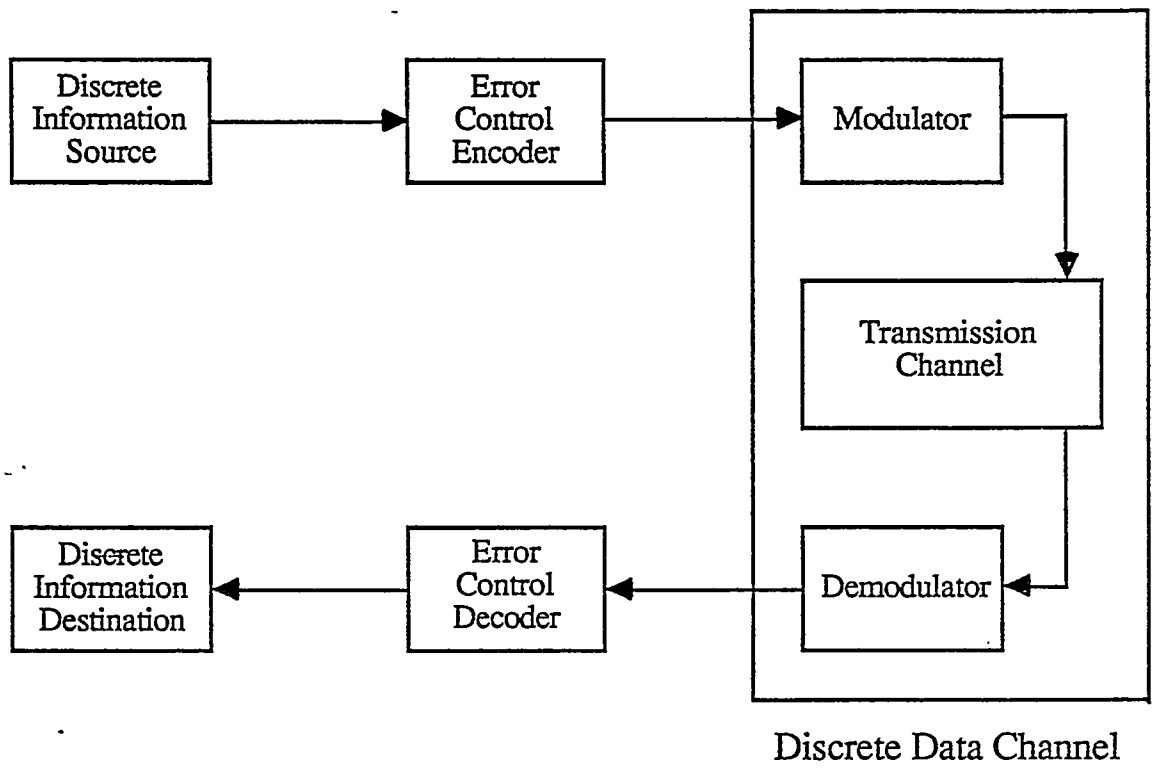

Figure 2-3: Simplified Communication System for Coding

The discrete data channel model is characterized by the input symbols, output symbols and the transition probabilities relating the outputs to each possible input. The transition probabilities represent the ways in which the signals are affected by amplitude and phase variations, equipment noise and nonlinearities, and channel noise. In most situations it is difficult to define a model that accounts for all the possible signal disturbances. In most coding literature, a simplified model called the discrete memoryless channel